

ESCOLA SENAI MORVAN FIGUEIREDO

Curso: Desenvolvimento de Sistemas – 2025\_DEV4\_SESI

Prof.: Johnny Braga de Oliveira

**TEMA – 3**

**Projeto Final –**

**🏥 Sistema Completo de Clínica Médica (Morvan HealthCare)**

**📌 Contexto**

Muitas clínicas ainda utilizam **papel ou sistemas fragmentados**, dificultando o gerenciamento de pacientes, agendas, prontuários e relatórios.

**⚡ Problemática**

O desafio dos alunos será criar um **sistema integrado**, onde pacientes, médicos, recepcionistas e administradores possam interagir de forma eficiente, com **agendamento online, prontuários digitais e relatórios completos**, simulando um ambiente de clínica real.

🛠️ **Arquitetura do Projeto**

**Back-end (Django)**

* Django (framework principal)
* Django REST Framework (API para app mobile e integrações externas)
* Django Allauth (autenticação e controle de usuários)
* Django Crispy Forms (formulários médicos aprimorados)
* Django Filter (busca avançada de pacientes, consultas e agendas)

**Banco de Dados**

* PostgreSQL (armazenamento de pacientes, médicos, consultas, prontuários e relatórios)

**Front-end**

* HTML, CSS, JavaScript
* Bootstrap ou Tailwind para interface responsiva
* Dashboards interativos com pandas + matplotlib

**Módulos Principais**

1. **Gestão de Usuários**
   * Perfis: paciente, médico, recepcionista, administrador
   * Autenticação e permissões avançadas (django-allauth)
2. **Agendamento Online**
   * Consulta de disponibilidade de médicos
   * Confirmação automática de agendamentos
   * Cancelamentos e reorganização de filas
3. **Área do Paciente**
   * Histórico de consultas
   * Emissão de receitas e documentos em PDF (reportlab)
   * Prontuário eletrônico com dados clínicos
4. **Área do Médico**
   * Agenda diária
   * Registro de evolução clínica
   * Emissão de laudos e receitas em PDF
5. **Área da Recepção**
   * Organização de filas e horários
   * Gerenciamento de cancelamentos e remarcações
6. **Relatórios e Dashboards**
   * Estatísticas sobre número de consultas, especialidades mais procuradas
   * Gráficos e tabelas usando pandas + matplotlib
7. **API REST**
   * Endpoints para integração com app mobile
   * Consulta de agenda, pacientes, laudos e relatórios

📦 **Bibliotecas Necessárias**

| Biblioteca | Instalação | Função |
| --- | --- | --- |
| django | pip install django | Framework principal |
| djangorestframework | pip install djangorestframework | Criação de API REST |
| reportlab | pip install reportlab | Geração de receitas, laudos e relatórios em PDF |
| django-crispy-forms | pip install django-crispy-forms | Formulários médicos personalizados |
| django-filter | pip install django-filter | Filtros avançados de pacientes, consultas e agendas |
| django-allauth | pip install django-allauth | Autenticação e controle de usuários |
| pandas | pip install pandas | Manipulação de dados para relatórios e dashboards |
| matplotlib | pip install matplotlib | Criação de gráficos analíticos |
| psycopg2 | pip install psycopg2 | Conexão com PostgreSQL |

📑 **Entregáveis**

1. **Prototipação UI/UX (Figma ou similar)**
   * Fluxos para paciente, médico, recepção e administrador
   * Dashboards interativos e protótipo navegável
2. **Documentação Técnica e Funcional**
   * Requisitos funcionais e não funcionais
   * Diagrama de entidades (DER)
   * Casos de uso (agendamento, emissão de receitas, visualização de prontuários)
   * Manual de instalação e configuração
   * Guia do usuário final
3. **Projeto Django Funcional**
   * CRUD completo de pacientes, médicos e consultas
   * Agendamento online com confirmação automática
   * Prontuário digital e emissão de PDFs (reportlab)
   * Dashboards com gráficos (pandas + matplotlib)
   * API REST funcional
4. **Apresentação Final (dezembro/2025)**
   * Demonstração do sistema completo
   * Protótipo navegável e dashboards interativos
   * Justificativa técnica das bibliotecas utilizadas
   * Discussão dos desafios enfrentados e soluções implementadas

**Aluno(a): Data:**

**Aluno(a): Data:**

**Aluno(a): Data:**

**Aluno(a): Data:**

**Aluno(a): Data:**